

AMENDMENT TO THE CLAIMS

12. (Currently Amended) An operand memory stack for use in a calculating machine containing a processing unit for processing individual operands according to a program, the operand memory stack comprising: ~~and the operand stack in which~~ operands of different lengths that are stored as a stack [[,]] ; and ~~eharaacterized~~ by

a type memory (20, 31) ~~with~~ comprising memory elements of constant length that store, ~~which stores~~ for each operand stored in the operand memory stack (10, 32) its , a type information that is coded and which contains information about the length of ~~the relevant~~ a corresponding operand [[,]] ;

wherein the length of ~~the~~ a particular operand type is being stored in a table in dependence on the corresponding coded type information ~~code~~.

13. (Currently Amended) An operand memory stack according to claim 12, ~~characterized in that~~ wherein the type memory (20) is formed separate from the operand memory stack as a stack with constant length stack elements ~~separate from the operand memory~~.

14. (Currently Amended) An operand memory stack according to claim 12, ~~characterized in that~~ wherein the type memory (31) is integrated ~~operand by operand~~ into the operand memory stack such that each operand is directly contiguous to the corresponding coded type information.

15. (Currently Amended) An operand memory stack according to claim 12, ~~characterized in that~~ wherein the operand memory stack is formed as a virtual stack for a virtual calculating machine.

16. (Currently Amended) An operand memory stack according to claims 12, 13, 14 or 15, further comprising ~~characterized by~~ an operand type checking device (S12-S14) which is activated at each read access to the operand memory stack (10, 32).

17. (Currently Amended) A calculating machine comprising ~~having~~ an operand memory stack according to claims 12, 13 or 14.

18. (Currently Amended) An operand memory stack according to claims 12, 13 or 14, wherein the operand memory stack is contained within a [[A]] smart card having an integrated virtual calculating machine ~~according to claim 12.~~

19. (Currently Amended) A method for operating an operand memory stack in a calculating machine, the method comprising: ~~wherein the~~
providing stack elements of the operand memory stack that are used for storing operands of different length, ~~characterized in that ;~~

creating a type memory element (~~20a, 20b, 31~~) of uniform length ~~is created~~ for each operand in the operand memory stack (~~10, 32~~);

storing the coded type information ~~stored in a type memory element that~~ contains length information about the length of the each corresponding operand in the type memory elements [[,]]; and

evaluating said length information ~~is evaluated~~ at each access to the operand memory stack [[,]]; and

storing the length of ~~the a~~ particular operand type ~~being stored~~ in a table in dependence on the corresponding coded type information ~~code~~.

20. (Currently Amended) A method according to claim 19, ~~characterized in that~~ wherein the type memory elements are created in the form of a separate stack (~~20~~).

21. (Currently Amended) A method according to claim 19, ~~characterized in that~~ wherein the type memory elements (~~31~~) are stored contiguously with the corresponding operand memory stack element (~~32~~).

22. (Currently Amended) A method according to claims 19, 20 or 21, ~~characterized in that~~ wherein a type check is performed at each read access to the operand memory stack (~~10, 32~~).